Simulation-Driven Machine Learning Framework for Semi-Dynamic Project-Delay Prediction

Sahana Ramachandran, Siyona Behera, Varsha Giridharan

***Abstract*—Schedule overruns remain a persistent issue across large-scale projects, highlighting the challenge of modeling uncertainty and interdependent risks in complex project networks. Existing analytical methods such as CPM/PERT scheduling, fuzzy earned-value analysis, and conventional Monte Carlo simulation remain largely static and treat risks as independent, limiting their ability to capture cascading effects. This study introduces a simulation-driven machine-learning framework that combines CPM/PERT-based Monte Carlo simulation with supervised learning to enable semi-dynamic delay-risk prediction. The approach captures risk interactions and uncertainty distributions empirically, rather than through predefined fuzzy or grey equations. Three classifiers—Logistic Regression, Naïve Bayes, and Random Forest—are trained on 630 synthetically generated project networks from the RG30 dataset, and evaluated on a held-out test set to predict delayed completion. The predictive accuracy is then compared against two baseline analytical models from prior work, demonstrating the benefit of simulation-based feature engineering for early-stage schedule-delay forecasting. The results show that uncertainty-aware features derived from simulation improve early-stage predictive performance compared to classical CPM- and EVM-based indicators**

***Index Terms*—Project scheduling, Monte Carlo simulation, Bayesian networks, machine learning, delay prediction, risk interaction, uncertainty modeling, project performance forecasting, earned value management, CPM/PERT.**

I. INTRODUCTION

PROJECTS are inherently dynamic and uncertain systems that often lead to risks emerging as execution progresses. These risks may propagate through interdependent activities, amplifying disruptions and causing significant schedule and cost overruns. The inability to forecast such delays results in substantial losses, particularly in construction, software, and engineering projects. Existing analytical methods either struggle to capture the cascading nature of risks or perform poorly when project data are dynamic. Even recent machine-learning approaches often rely on static or subjective inputs, which limits their predictive accuracy. This study addresses these limitations by training machine-learning models on simulation-generated uncertainty features rather than static inputs, enabling early-stage delay prediction.

Conventional scheduling and risk-assessment techniques such as the Critical Path Method (CPM), Program Evaluation and Review Technique (PERT), Earned Value Management (EVM), and Monte Carlo (MC) simulation form the foundation of project-performance analysis. These frameworks quantify task dependencies, estimate expected durations under uncertainty, and evaluate progress through cost and schedule indices. However, they are largely static: each forecast is computed from fixed input parameters and must be recalculated manually as new information becomes available. As a result, such methods fail to capture evolving project conditions or interdependent risks that cascade through the network. Later extensions such as fuzzy EVM, grey-system estimation, and hybrid analytical models introduced probabilistic representations of uncertainty, yet they remain snapshot-based and non-adaptive.

Consequently, recent research has attempted to overcome the limitations of static risk-analysis by explicitly modelling risk interdependencies and uncertainty. For example, the PN–RIN model integrates a Project Network (PN) and a Risk Interaction Network (RIN) to capture how risks propagate through causal links and affect activity durations, and applies simulation-based control metrics to sensitive activities. In another stream, fuzzy earned-value analysis extends classical EVM by representing schedule and cost indices as fuzzy sets, thereby accounting for ambiguous estimates of performance metrics. Further, SEM-BN models combine structural equation modelling with Bayesian networks to forecast project outcomes under uncertainty; however, despite their sophistication these approaches rely on predefined causal structures or static data snapshots. Collectively, these studies demonstrate that while risk interaction and uncertainty modelling have become more mature, forecasting methods remain constrained by implicit or static assumptions. Hence, there remains a need for a framework that learns risk propagation patterns directly from data and adapts early-stage project signals into predictive models.

To address these remaining limitations, this paper proposes a simulation-driven machine learning framework for early stage project delay prediction. The framework integrates CPM/PERT-based Monte Carlo simulation with supervised learning to generate and analyze synthetic project data that capture structural, uncertainty, and early performance characteristics. Instead of relying on predefined fuzzy or causal equations, the proposed approach learns risk-propagation patterns empirically from simulation outcomes. Three classifiers—Logistic Regression, Naïve Bayes, and Random Forest—are trained on 630 synthetically generated project networks and evaluated on a held-out test set, using data similar to prior literature. The inclusion of early Earned Value Management (EVM) metrics such as the Schedule Performance Index (SPI) and Cost Performance Index (CPI) at 20% progress enables semi-dynamic forecasting that reflects evolving project conditions. Comparative experiments against an analytical baseline from prior work assess the predictive improvement introduced by simulation-based feature engineering.

The proposed framework offers a scalable and reproducible approach for combining simulation outputs with data-driven learning, providing project managers with a quantitative tool to anticipate schedule delays before they escalate. Although demonstrated here on synthetic datasets, the methodology can be extended to real-world project environments and evolved toward fully dynamic, real-time forecasting systems. The remainder of this paper is organized as follows: Section II reviews the literature; Section III describes the proposed methodology; Section IV presents experimental results and discussion; Section V explores the limitations; Section VI describes the Conclusions; and Section VII concludes the paper with directions for future work.

II. Literature Review

Project-delay prediction has been approached from multiple analytical and probabilistic perspectives. Past studies have attempted to capture uncertainty, interdependence, and causal relationships through different modelling methods. This section reviews three representative works that serve foundational to the present study and identifies the gaps that motivate the proposed framework.

Song and Vanhoucke proposed a two-layer Project Network–Risk Interaction Network (PN–RIN) model to analyse how project-level risks propagate through activity dependencies. The project network represents task precedence, while the risk network encodes interdependencies among risk factors such as resource unavailability, design errors, or procurement delays. The integration of both layers enables simulation of cascading failures across the network. The authors introduced a Risk Criticality Index (RCI) and a Risk Significance Index (RSI) to quantify each risk's contribution to overall delay. Their experiments demonstrated that the PN–RIN model captures indirect and compounding effects that traditional CPM or PERT overlook. Although PN–RIN successfully formalises risk interdependence, it remains an analytical framework that does not generalise across projects. Its parameters and causal links must be defined manually, and the model does not learn from empirical or simulated data. Hence, while it advances the representation of risk interactions, it remains static and descriptive rather than predictive.

Fan et al. advanced project-performance evaluation under uncertainty by integrating fuzzy set and grey system theories within an Earned Value Management (EVM) framework. Their Grey-Fuzzy EVM (GF-EVM) model expresses project progress using interval grey triangular fuzzy numbers (IGTFNs) that simultaneously capture linguistic vagueness and confidence intervals. By quantifying expert-based assessments of cost and schedule performance, the approach provides a more realistic representation of uncertainty than classical EVM. The authors validated the model on a three-phase industrial project and compared its performance with Z-number EVM and traditional fuzzy EVM, reporting improved cost- and schedule-forecast accuracy. However, the model's uncertainty representation remains subjective and snapshot-based. Both fuzzy and grey parameters are pre-defined by experts and do not evolve with incoming progress data. Consequently, while GF-EVM enhances interpretability, it lacks the adaptivity required for dynamic or data-driven forecasting. The present study builds on this limitation by generating uncertainty features through Monte Carlo simulation rather than expert definition, allowing supervised-learning models to empirically learn uncertainty patterns from data.

More recent studies by Unsal-Altuncan and Vanhoucke combine Structural Equation Modelling (SEM) with Bayesian Networks (BN) to capture both causal relationships and probabilistic dependencies among project variables. The SEM component quantifies latent factors such as management efficiency or communication quality, while the BN component infers conditional probabilities of delay or cost overrun. The hybrid SEM–BN model was trained on synthetically generated projects using RanGen2 (a random project generator) and compared with traditional Monte Carlo forecasts and a few well known ML models. Results indicated improved predictive accuracy for time and cost performance relative to purely analytical baselines. Despite this improvement, the SEM–BN framework still relies on predefined causal structures and simulated static datasets. The model parameters are calibrated once and do not adapt as project progress data become available. Its so-called "dynamic" simulation merely reproduces progress snapshots through a second Monte Carlo run rather than learning from evolving data. Thus, while it provides a stronger probabilistic foundation than earlier analytical models, it remains pseudo-dynamic and limited in generalisability.

Collectively, the PN–RIN, fuzzy EVM, and SEM–BN approaches advance the understanding of project-risk interaction and uncertainty quantification. Yet all three share two fundamental limitations: (1) their reliance on predefined structures or expert judgment, and (2) their static treatment of project data. None of the models adaptively learn from empirical or simulated progress information.

The present study addresses this gap by employing Monte Carlo simulation as a data generator rather than merely a baseline forecaster and by training machine-learning classifiers on early-stage performance features, including SPI and CPI. This simulation-driven learning paradigm enables semi-dynamic delay prediction and establishes a bridge between analytical project-risk models and modern data-driven forecasting methods.

III. Methodology

A. Novelty

This study introduces a simulation-driven data-generation pipeline that differs fundamentally from static analytical methods and expert-defined fuzzy or grey methods. The novelty lies in extracting empirical uncertainty behaviour from project networks, which shifts the core idea from representing uncertainty to learning it. This study uses artificial project data from RanGen2 RG30 (Vanhoucke et al., 2008) to ensure compatibility with established benchmarks. However, instead of using the project data as is, PERT triplets are created around the durations to introduce uncertainty. The data is then subjected to CPM and 200-run Monte Carlo simulations from which distributional features—such as mean and variance of activity durations, critical-path sensitivity, probability of late completion, and indicators of network-level uncertainty—are extracted. Hence, this study produces simulation-derived uncertainty features that implicitly encode risk interactions without requiring expert rules, linguistic uncertainty scales, or predefined causal links. Additionally, early EVM signals including Schedule Performance Index (SPI) and Cost Performance Index (CPI) are computed at 20% progress, enabling forecasting that partially reflects evolving project reality. This introduces semi-dynamic prediction, unlike snapshot-based methods in prior work.

Machine-learning models then learn patterns embedded in the simulation data. This allows risk interactions to emerge naturally through Monte Carlo sampling with no need for explicit RIN or SEM-BN structures. This also helps avoid reliance on linguistic fuzzy sets or expert-defined grey intervals. To contextualize the benefit of simulation-derived features, the predictive performance of the machine-learning models is compared against a simple analytical baseline, where delay classification is derived directly from deterministic CPM estimates or the Monte Carlo late-finish probability.

Collectively, these contributions establish a data-driven, generalizable, and causal-structure-free framework for project-delay prediction that bridges classical project-risk analysis with modern machine-learning methodologies.

B. Phase 1: Data Generation

The project data was obtained from the RG30 dataset. Out of five sets in the folder, only the first set with 900 instances (where each instance one project) was used to match the experimental setup in prior work by Unsal-Altuncan and Vanhoucke. As the projects were in Patterson format, a custom parser was built to convert each project into a Directed Acyclic Graph (DAG) and extract the predecessors of each task.

To introduce duration uncertainty, PERT triplets of the form [optimistic estimate (o), most-likely estimate (m), pessimistic estimate (p)] were assigned to each activity. The original RG30 durations were used as the most-likely duration and ‘o’ and ‘p’ were computed using the formulae

$$\begin{aligned} o &= \max(1, \mathrm{round}(0.7 \cdot m)) \\ p &= \max(m+1, \mathrm{round}(1.4 \cdot m)) \end{aligned} \tag{1}$$

The 0.7m and 1.4m factors provide a realistic but controlled spread around the most-likely estimate while preserving the positive skew expected in construction and engineering activity durations. This also ensures comparability with prior literature, where uncertainty is introduced through fixed perturbation ranges rather than expert-defined fuzzy or grey intervals.

Using these values, a baseline schedule was calculated by running an iteration of Critical Path Method (CPM) forward pass and backtracking. A small buffer factor of 1% was added to emulate mildly pessimistic-leaning conditions.

Early progress aggregates—Actual Cost (AC), Planned Value (PV), Earned Value (EV)—were then calculated by running Earned Value update on baseline metrics at 20% progress. Since the RG30 dataset does not include cost data, time-based proxies were used as they are positively correlated with both schedule and cost growth (PMBOK Guide, Project Management Institute).

To estimate probability of delay and other uncertainty metrics, 200 runs of Monte Carlo simulation were performed for each project using its PERT distributions. Each run sampled new activity durations, recomputed CPM finish time and compared it with the baseline. The proportion of runs exceeding the baseline finish time was recorded as plate. A project was labeled delayed if plate was greater than 0.5, on-time otherwise.

This process produced a fully synthetic dataset that preserves network structure from RG30 while also introducing uncertainty and early-progress information required for performing supervised learning.

C. Phase 2: Feature Engineering

After generating the synthetic project data and Monte Carlo outcomes, the next step involved constructing features suitable for supervised learning.The features used for this experiment can be broadly divided into three categories—structural, uncertainty-based, and early progress.

Structural features—number of tasks, number of edges, network density, critical path length, percentage of critical tasks, baseline total duration—describe the topology and baseline logic per project. These features capture how structural complexity influences schedule sensitivity.

Uncertainty-based features—mean most likely duration, mean range of pessimistic and optimistic estimates, instability index—summarize the uncertainty distributions emerging from Monte Carlo sampling. The range of duration for each task is defined as p – o, the mean was calculated to get an idea of the uncertainty spread. Similarly, instability is defined as the variance of the total durations calculated in the 200 Monte Carlo runs. These features approximate how duration uncertainty propagates through the network.

Early progress features—Cost Performance Index (CPI), Schedule Performance Index (SPI)—provide semi-dynamic signals reflecting early project performance. These are defined by the early performance metrics Actual Cost (AC), Earned Value (EV), and Planned Value (PV) calculated at 20% progress. CPI is calculated as EV/AC, and SPI as EV/PV. These indicators provide an early snapshot of project health and complement the structural and uncertainty-driven predictors.

D. Phase 3: Preprocessing

Before training ML models on the data, preprocessing techniques were applied to prevent data leakage and consistent feature scaling.

To address potential leakage, columns p_late_diag, buffer factor, and n_tasks were dropped. The first is directly derived from the target variable which would inflate results, and the latter two are constant (0.01, 30 respectively) and don’t add any useful predictive signal. Similarly, the CPI and SPI values were clipped to a range of [0, 2] to prevent outliers from disproportionately influencing the data.

This data is then split into input features X and target variable y. A stratified 70–15–15 train/validation/test split was used to preserve the class distribution across all sets. A scaled version of inputs was created for logistic regression as scaling leads to improved convergence for linear models. Random Forest and Naïve Bayes were trained on the unscaled features.

E. Phase 4: Model Selection

Three supervised learning models were chosen, each representing a linear, non-linear, or generative decision boundary: Logistic Regression, Random Forest, Naïve Bayes. This combination allows us to benchmark and evaluate whether simulation-derived uncertainty features offer predictive value compared to traditional analytical logic.

Logistic Regression was chosen as a linear baseline model to evaluate whether simple relationships among structural and uncertainty features are sufficient for delay prediction. The model was trained on scaled features with default hyperparameters and L2 regularization.

The second model used was Gaussian Naïve Bayes, which models likelihood of each feature under each class using a normal distribution and combines them through Bayes' rule. It was chosen as a generative baseline that tests whether engineered features provide separability under strong independence assumptions. The model was trained on unscaled features with default hyperparameters.

The primary non-linear model selected was Random Forest due to its ability to capture effects of interaction among the three categories of features. It was trained on unscaled features using hyperparameters n_estimators = 300, max_depth = 5, and min_samples_leaf = 5 as tuned on the validation set.

Together, these models enable comparison across linear, probabilistic, and nonlinear approaches and help assess whether simulation-derived features improve early-stage delay prediction relative to classical analytics.

F. Phase 5: Analytical Baselines

This study defines two analytical baselines to evaluate and further contextualize the performance of machine-learning models. These reflect traditional project-management logic and are aligned with methods used in prior literature, including EVM and structural CPM-based analysis.

A simple EVM rule was used as the first analytical baseline. The Schedule Performance Index , as computed in Phase 2, was compared against a threshold of 1 to identify delay. As the ratio is centred around 1, projects with SPI $\leq$ 1 were classified as delayed. This heuristic tends to be unstable during early stages, since EV and PV values are highly sensitive to initial duration estimates.

The second baseline was a logistic regression classifier trained only on structural CPM features— number of edges, network density, critical path length, percentage of critical tasks, and baseline project duration. This captures purely deterministic structural logic and mirrors the role of variables used in analytical methods such as CPM, PN–RIN, and SEM–BN. It excludes all uncertainty-related and EVM-derived features, allowing us to evaluate whether structural information alone can predict schedule delay. Together, these baselines represent traditional deterministic forecasting approaches and enable direct comparison with the simulation-driven machine-learning models developed in this study.

TABLE I
COMPARING PERFORMANCE OF LR, NB, RF ON VALIDATION TEST

Model	**Accuracy**	**F1 Score**	**ROC-AUC**
Logistic Regression	0.69	0.71	0.74
Naive Bayes	0.68	0.48	0.5
Random Forest	0.73	0.75	0.74

TABLE II
COMPARING RANDOM FOREST WITH THE TWO ANALYTICAL BASELINES

Method	Accuracy	F1 Score	ROC-AUC
Early SPI	0.496	0.261	0.504
Structural Inference: Logistic Regression	0.681	0.699	0.748
Random Forest	0.689	0.700	0.758

IV. RESULTS

Experiments were conducted on 900 synthetic project instances from the RG30 dataset using a stratified 70-15-15 train/validation/test split. Models were selected based on validation performance using accuracy, F1-score, and ROC-AUC, with the test set held out solely for final evaluation.

Table I compares validation performance of Logistic Regression (LR), Gaussian Naive Bayes (NB), and Random Forest (RF), Naive Bayes performs the weakest, reflecting its independence assumptions and sensitivity to feature distributions. Logistic Regression achieves moderate performance, showing that linear relationships capture some delay patterns but miss nonlinear effects. Random Forest achieves the hughes F1 and ROC-AUC scores on the validation set, and was thus selected as the primary model for testing and baseline comparison.

Table II and Fig. 1 compare RF's performance against the two analytical baselines. Baseline A (early SPI) performs near random (ROC-AUC of 0.50), confirming its instability at 20% progress. Baseline B (structural logistic regression) achieves a much better performance (ROC-AUC of 0.75), in line with prior RIN/SEM literature, emphasizing the importance of network topology. The tuned RF model achieved an ROC-AUC of 0.76 on the held-out test set, with an accuracy of 0.69 and F1-score of 0.70, indicating that simulation-driven uncertainty features and nonlinear interactions offer higher predictive value, more than deterministic CPM structure.

Feature importance analysis (Fig. 2) show that critical-path structure and early CPI are the strongest predictors of delay. Uncertainty-derived features like instability_m and mean_range_po contribute secondary effects, while SPI contributes minimal importance.

Shapely Additive Explanation (SHAP) summary bar plot (Fig. 3) ranks features based on their average absolute contribution to model output. This further reiterates that critical-path structure and CPI are the strongest global predictors of delay. The next biggest contributors are uncertainty-driven variables, confirming that Monte Carlo-derived features add predictive

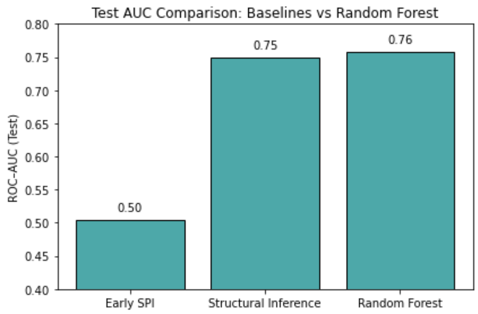

Fig. 1. Bar chart comparing ROC-AUC for Analytical Baselines and Random Forest

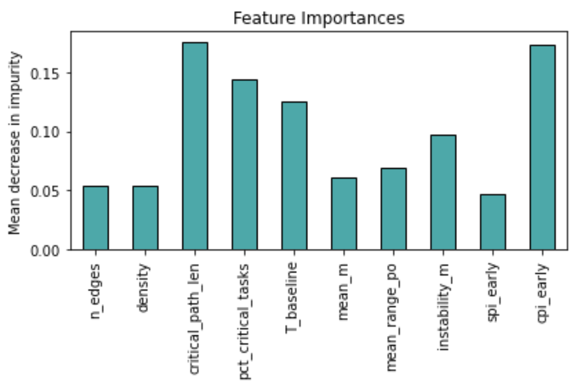

Fig. 2. Mean decrease in impurity in features

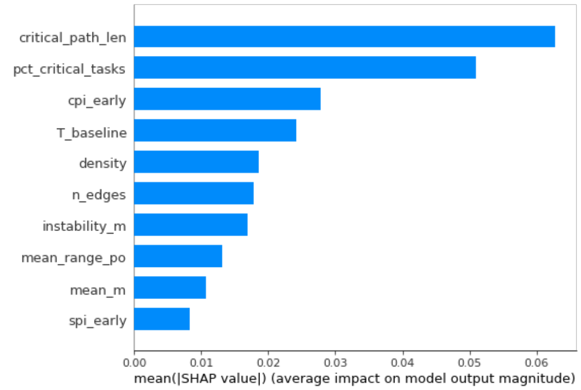

Fig. 3. Mean SHAP values

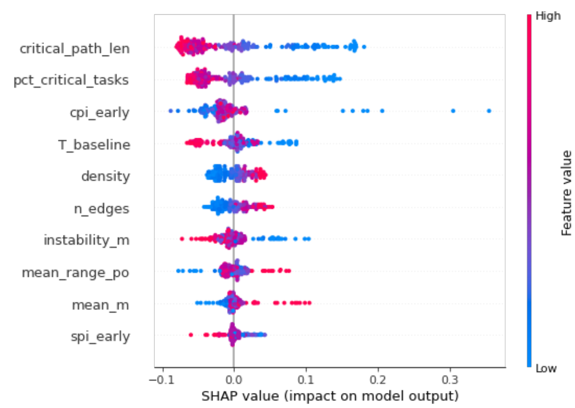

Fig. 4. Impact of features on model output

value beyond CPM topology. These ranking closely align with RF's impurity-based importance scores.

The SHAP beeswarm plot (Fig. 4) shows us how feature values influence predictions. Projects with longer critical-path lengths tend to exhibit positive SHAP values, pushing predictions toward the 'delayed' class. Low early CPI strongly increases predicted delay risk, whereas high early CPI shifts prediction towards the 'on-time' class. Uncertainty features show mixed directional effects, indicating that their impact depends on interactions with structural variables. This highlights the nonlinear, interaction-driven behavior that RF captures but linear baselines cannot.

Overall, these results show that combining structural CPM information with uncertainty-aware features and early EVM metrics provide the most reliable early-stage delay predictions.

V. Limitation

Despite promising results, this study has several limitations. First, the experiments rely exclusively on synthetic project networks from the RG30 dataset, which limits external validity; real-world project data may exhibit more complex patterns, irregular structures, and non-PERT uncertainty distributions. Second, early-progress indicators were approximated using time-based cost proxies due to the absence of cost data in RG30, which may affect the accuracy of CPI- and SPI-based features. Third, the Monte Carlo simulation depth (200 runs) and PERT perturbation factors were fixed uniformly across projects, which may oversimplify how uncertainty behaves in real environments. Finally, the models were evaluated using static early-progress snapshots at 20

VI. Conclusions

This study developed a simulation-driven framework for early-stage delay prediction by integrating CPM/PERT-based Monte Carlo simulation with supervised learning. Using 900 RG30 project networks, the proposed approach generated uncertainty-aware and early-progress features that capture both deterministic project topology and dynamic risk propagation. Experimental results show that relying solely on early EVM indicators such as SPI is insufficient for reliable delay classification, and while structural CPM features provide a strong baseline, Random Forest achieved the highest predictive performance (ROC-AUC $\approx$ 0.76). This indicates that uncertainty ranges, instability, and early CPI provide incremental value beyond deterministic topology. SHAP analysis further demonstrated that both structural logic and uncertainty effects shape early delay risk. Overall, the results suggest that simulation-derived uncertainty metrics can enhance traditional analytical models and offer a scalable, data-driven approach for early-stage forecasting. This framework can be extended to real project datasets and adapted toward more dynamic, continuous-update prediction systems.

VII. Future Work

Several extensions can strengthen and generalize the proposed framework. First, applying the methodology to real project datasets would validate its external applicability and allow calibration of uncertainty distributions using empirical duration data rather than fixed PERT factors. Second, incorporating more dynamic weekly EV updates or activity-level variance would support more dynamic and fine-grained forecasting. Third, the simulation engine can be expanded to include resource-driven delays, enabling more realistic modelling of risk propagation. Finally, additional interpretability techniques—such as partial dependence plots or causal sensitivity analysis—could further illuminate how structural and uncertainty features contribute to early delay risk.

References

[1] Ünsal-Altuncan, İ., & Vanhoucke, M. (2024). A hybrid forecasting model to predict the duration and cost performance of projects with Bayesian Networks. *European Journal of Operational Research*, 315(2), 511–527. https://doi.org/10.1016/j.ejor.2023.12.029

[2] Fan, L., Mohseni Nejad, S., Bagherpour, M., Feylizadeh, M. R., & Karimi, N. (2025). Modeling sustainable earned value management (EVM) under grey uncertain conditions. *Systems*, 13(6), 484. https://doi.org/10.3390/systems13060484

[3] Song, Y., & Vanhoucke, M. (2025). Schedule risk analysis for project control with risk interactions. *Annals of Operations Research*. https://doi.org/10.1007/s10479-025-06668-8

[4] Ghent University, "Project Data" [Online]. Available: https://www.projectmanagement.ugent.be/research/data

[5] Vanhoucke, M., & Coelho, J. (n.d.). RanGen: Random project network generator. *Project Management Research Group, Ghent University.* Available: https://www.projectmanagement.ugent.be/research/data/RanGen

[6] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. Available: http://jmlr.org/papers/v12/pedregosa11a.html

[7] Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

[8] Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems 30* (NIPS 2017), 4765–4774.